



The
Patent
Office

RECD 04 MAY 1999

WIPO

PCT

PCT/GB 99/00719



INVESTOR IN PEOPLE

5

PRIORITY DOCUMENT

SUBMITTED OR TRANSMITTED IN
COMPLIANCE WITH RULE 17.1(a) OR (b)

The Patent Office
Concept House
Cardiff Road
Newport
South Wales
NP9 1RH

23/9

GB99/719

I, the undersigned, being an officer duly authorised in accordance with Section 74(1) and (4) of the Deregulation & Contracting Out Act 1994, to sign and issue certificates on behalf of the Comptroller-General, hereby certify that annexed hereto is a true copy of the documents as originally filed in connection with the patent application identified therein.

I also certify that by virtue of an assignment registered under the Patents Act 1977, the application is now proceeding in the name as substituted.

In accordance with the Patents (Companies Re-registration) Rules 1982, if a company named in this certificate and any accompanying documents has re-registered under the Companies Act 1980 with the same name as that with which it was registered immediately before re-registration save for the substitution as, or inclusion as, the last part of the name of the words "public limited company" or their equivalents in Welsh, references to the name of the company in this certificate and any accompanying documents shall be treated as references to the name with which it is so re-registered.

In accordance with the rules, the words "public limited company" may be replaced by p.l.c., plc, P.L.C. or PLC.

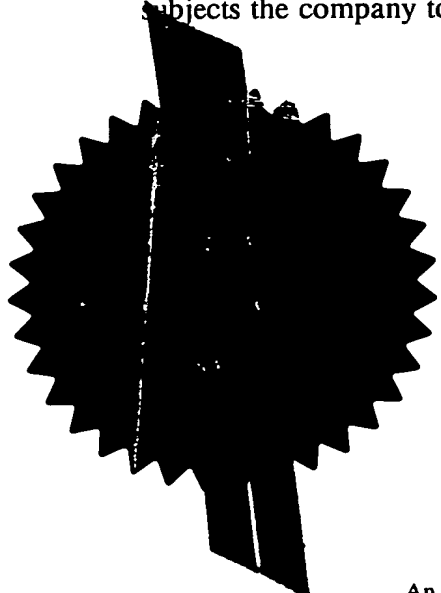
Re-registration under the Companies Act does not constitute a new legal entity but merely subjects the company to certain additional company law rules.

Signed

G D Court.

Dated

08 APR 1999



This Page Blank (uspto,

13MAR98 E345055-1 006256
P01/7700 25.00 769243.4
The Patent Office

Request for grant of a patent

(See the notes on the back of this form. You can also get an explanatory leaflet from the Patent Office to help you fill in this form)

Cardiff Road
Newport
Gwent NP9 1RH

1. Your reference **P/61380**

THE PATENT OFFICE
B
13 MAR 1998

2. Patent application number
(The Patent Office will fill in this part)

8661 NAW 51
13 MAR 1998
RECEIVED BY POST

9805243.4

3. Full name, address and postcode of the or of each applicant (underline all surnames)

GPT LIMITED
CENTURY PARK
P.O. BOX 53
COVENTRY
CV3 1HJ

Patents ADP number (if you know it)

722020500

If the applicant is a corporate body, give the country/state of its incorporation

ENGLAND

APPLICATION FILED 18.01.99

4. Title of the invention

BROADBAND SERVICE CREATION ENVIRONMENT

5. Name of your agent (if you have one)

H A BRANFIELD

"Address for service" in the United Kingdom to which all correspondence should be sent (including the postcode)

**GEC PATENT DEPARTMENT
WATERHOUSE LANE
CHELMSFORD
ESSEX CM1 2QX**

Patents ADP number (if you know it)

5858519002 ✓

6. If you are declaring priority from one or more earlier patent applications, give the country and the date of filing of the or of each of these earlier applications and (if you know it) the or each application number

Country

Priority application number
(if you know it)

Date of filing
(day / month / year)

7. If this application is divided or otherwise derived from an earlier UK application, give the number and the filing date of the earlier application

Number of earlier application

Date of filing
(day / month / year)

8. Is a statement of inventorship and of right to grant of a patent required in support of this request? (Answer 'Yes' if:

YES

- a) any applicant named in part 3 is not an inventor, or
 - b) there is an inventor who is not named as an applicant, or
 - c) any named applicant is a corporate body
- See note (d))

Patents Form 1/77

9. Enter the number of sheets for any of the following items you are filing with this form.
Do not count copies of the same document

Continuation sheets of this form -

Description 14

Claim(s) 1

Abstract -

Drawing(s) 7 + 7

10. If you are also filing any of the following, state how many against each item.

Priority documents -

Translations of priority documents -

Statement of inventorship and right to grant of a patent (Patents Form 7/77) -

Request for preliminary examination and search (Patents Form 9/77) 1

Request for substantive examination (Patents Form 10/77) 1

Any other documents (please specify) -

11. I/We request the grant of a patent on the basis of this application.

Signature
H A BRANFIELD

Date 12.03.98

12. Name and daytime telephone number of person to contact in the United Kingdom

H A BRANFIELD

01245 275132

Warning

After an application for a patent has been filed, the Comptroller of the Patent Office will consider whether publication or communication of the invention should be prohibited or restricted under Section 22 of the Patents Act 1977. You will be informed if it is necessary to prohibit or restrict your invention in this way. Furthermore, if you live in the United Kingdom, Section 23 of the Patents Act 1977 stops you from applying for a patent abroad without first getting written permission from the Patent Office unless an application has been filed at least 6 weeks beforehand in the United Kingdom for a patent for the same invention and either no direction prohibiting publication or communication has been given, or any such direction has been revoked.

Notes

- If you need help to fill in this form or you have any questions, please contact the Patent Office on 0645 500505.
- Write your answers in capital letters using black ink or you may type them.
- If there is not enough space for all the relevant details on any part of this form, please continue on a separate sheet of paper and write "see continuation sheet" in the relevant part(s). Any continuation sheet should be attached to this form.
- If you have answered 'Yes' Patents Form 7/77 will need to be filed.
- Once you have filled in the form you must remember to sign and date it.
- For details of the fee and ways to pay please contact the Patent Office.



The
Patent
Office



INVESTOR IN PEOPLE

GB9805243.4

By virtue of a direction given under Section 30 of the Patents Act 1977, the application is proceeding in the name of

MARCONI COMMUNICATIONS LIMITED,
Incorporated in the United Kingdom,
New Century Park,
PO Box 53,
COVENTRY,
CV3 1HJ,
United Kingdom

[ADP No. 07519200001]

This Page Blank (uspto)

BROADBAND SERVICE CREATION ENVIRONMENT**1. Introduction**

A graphical service creation environment capable of creating Intelligent Network (IN) services for an integrated Broadband-Integrated Services Digital Network/Intelligent Network (B-ISDN/IN) network is described. This broadband service creation environment is based on an extension of an equivalent narrowband product such as GAIN INventor supplied by GPT Limited. GAIN INventor is one of a family of products based on the UNIXTM operating system and a Service Logic Execution Environment (SLEE). The SLEE is designed to support telecommunication services Service Logic Programs and is based on the SLEE API defined in the Bellcore AIN 1.0 standard. As well as residing on the INventor, the SLEE resides on the target environment, the Service Control Point (SCP). One of INventor's features is the graphical service creation environment which allows the creation of telecommunication services using graphical icons connected together to form a service picture. The service picture is a general logical description of a service without any explicit reference to the Service Control Function (SCF/SSF) or Service Switching Function (SRF) interface messages other than those at a higher level.

Section 2 is provided as background material. The approach within the project has been to provide switched broadband services which are extensions to the classical IN architecture as this does not dramatically change the typical service deployment architecture. The Service Switching Points (SSP), Service Control Points (SCPs) and Intelligent Peripherals (IP) have been enhanced to cope with the broadband

functionality. The broadband SCP (B-SCP) is realised as enhancements to an existing narrowband product. Service logic programs required to provide the complex broadband services to run on the B-SCP have been created using a broadband graphical environment (B-SCE).

5

According to the present invention a broadband graphical service creation environment for creating Intelligent Network services for an Integrated Broadband Service Digital Network/Intelligent Network network, comprising building blocks, a picture editor and code generator, the building blocks being a collection of broadband specific actions, data access and data manipulation routines plus pictorial blocks that define the graphical layout of the service.

10

The present invention will now be described, by way of example, with reference to the accompanying drawings, in which:-

15

Figure 1 shows the Entity Relationship Diagram for the Switching State Model;

Figure 2 shows the State Diagram for the Legs of Figure 1;

Figure 3 shows the State Diagram for the Connection of Figure 1;

Figure 4 shows the Finite State Machine for Broadband Service Control Function;

20

Figure 5 shows the Message Sequence for the *AddBearer* Option in the *BearerControl* Generic Service Building Blocks;

Figures 6A and 6B combined show the Runtime Action of the *AddBearer* Option in the *BearerControl* GSBB;

Figure 7 shows a table of GSBBs; and

Figure 7 shows a Service Creation Architecture.

2. IN based broadband services

The most significant impacts of a broadband network on IN with respect to service creation are:

- the number of B-ISDN calls that can comprise an IN service are now numerous;
- in dealing with the events relating to the multiple calls, the sequence of arrival of messages at the service logic can be unpredictable;
- the interaction between the network and user that arises from broadband services is of increased complexity.

Solutions to these problems include introducing a switching state model in the Service Switching Function, a finite state machine in the Service Control Function that copes with the asynchronous messages, and by introducing a “very intelligent” Intelligent Peripheral to handle the complex user interaction.

2.1 Switching State Model

The Switching State Model (SSM) is a function within the B-SSP that has the task of co-ordinating the events and detection points (DPs) from the several Basic Call State Models that form a service session. It offers an object-oriented view of the network elements and resources of the call control function (CCF) to the service logic. This object model removes the complexity at the B-SCP to co-ordinate the DPs at the service

logic. The model is capable of representing both abstract attributes like ownership of the session, as well as more concrete elements of the call like parties and connections.

Figure 1 shows the entity relationship diagram for the switching state model.

5 Within this model several parties can join a session, where a session is a representation of a complete call configuration as seen by an IN service, and a party represents real end users or network nodes like the B-IP or even the B-SCP. Between such parties, connections can be established, where connections can either be bearer related or bearer unrelated. A connection is composed of several legs, where the leg represents the
10 communication path to a party which is connected to other parties by a connection. A significant attribute of connections and legs is the status. This represents the status of the respective object in the call processing. Figures 2 and 3 show the state diagrams for the bearer and leg objects. Changes in the status attribute of legs and connections can be reported to the service logic when a state change takes place.

15

2.2 The Broadband INAP

The information flows between the B-SCF and the B-SSF are based on the abstract representation of the SSM model. The B-SCF is only able to manipulate the objects in the model, and as a result the operations for the information flows only allow such
20 object manipulation.

The protocol between the B-SSP and B-SCP that has emerged is based on the manipulation of the objects represented in the SSM model. Verbs such as *Add* and *Delete* are applied to objects such as parties and connection to provide this protocol. For

example if the service logic wishes to add a new bearer connection between two parties in the existing IN session, the operation *AddBearerToSession* is used to instruct the B-SSP to perform this operation. This has led to a Broadband IN Application Protocol (B-INAP) that is different from the current IN-CS 1 or 2 standard but has similarities with the emerging IN-CS3.2 standard. The B-SCP maintains its own view of the SSM model as relationship information is transmitted in the information flows. The model in the B-SCF is aligned with the model in the B-SSP which means that if an event changing the SSM state occurs, a communication flow is used to keep the SSM state aligned in the two nodes. State changes occur as a result of an event notified by the B-CCF, in which case B-SSF to B-SCF information flow takes place. The control as to which state changes are reported to the service logic is under the control of the service logic itself. The service logic must send requests to the B-SSP so that the particular state changes that occur are reported. For example, in the *AddBearerToSession* operation mentioned above, in order for the service logic to receive notification that the bearer connection has been set up, it must send a *RequestReportSSMChange*, specifying the transition *BeingSetUp* to *SetUp* prior to sending the message. Thus when the network connection is actually set up, the service logic is informed about the establishment of the connection via an operation *ReportSSMChange* that indicates that the bearer status has changed to the state *SetUp*.

2.3 B-SCF finite state machine

A significant consequence of the SSM and the B-INAP as far as the B-SCP is concerned is the finite state machine on the access side. After the receipt of the message which triggers the service logic, the *ServiceRequest*, the B-SCP has to explicitly arm the

necessary SSM state changes in order to keep track of the status of the calls and/or connections. The service logic needs to be informed when connections are released for any reason, and as this is achieved by means of a *ReportSSMChange* message, the service logic must be in a position to accept such a message in any state after the IN service logic is triggered. Figure 4 shows the finite state machine that the B-SCP access manager must be able to cope with State A represents the idle state where the service logic is waiting for the initial IN trigger (*ServiceRequest*). With this event the state changes to State B. This is an all encompassing state where messages are prepared for sending to the B-SSP but importantly it must also be able to accept messages from the B-SSP during this state. This is due to the fact that calls and/or connections can be released at any time which results in a notification to the service logic.

2.4 User Interaction

In the services considered, end users using a set top box can enter an interactive phase of the service, or navigation which allows the user to enter passwords, update service profiles or make selections that influence the IN service logic. As each service has different requirements, two approaches to user interaction were developed. For simple and generic interaction which is intended to work together with any end system, the feature User Service Interaction (USI) has been introduced. This uses a signalling connection between the end system and the B-SCP to convey user service information. For complex multimedia interaction, a broadband Specialised Resource Function (B-SRF) that supports GUI based interaction with the end system has been introduced.

Connection Oriented Bearer Independent (COBI) has been adopted as a transport means

to allow direct interaction between user and service logic. The handling of call related events and call unrelated events are presented in a unified view to the B-SCP by the IN-SSM model. The B-SRF provides the same functionality as the narrowband case with the main exceptions that it consists of its own logic and processing capability to work together with the broadband CPE in providing the services in a user friendly way and that it interacts with the network through a specialised interface to the B-SCF. The creation of service logic that resides on the B-SRF is not considered but tools for creating the service logic that reside and are executed on the B-SCF are considered.

3. A Service Creation Environment for broadband IN services

The narrowband service creation environment is based on the logical connections of graphical building blocks, referred to as *Generic Service Building Blocks* (or GSBBs) herein, however the concepts can be equally applied to a broadband environment. The GSBBs are an approach to the *Service Independent Block*, or SIB concepts in IN-CS-1.

The SCE consists of a finite number of GSBBs that can be selected and customised in a GSBB. Customised instances of the GSBBs or *icons*, are placed on a sketcher and linked together to form the service logic flow required into a picture file. A code generator converts the picture file into run time files required to run the service. There are broadly speaking three classes of GSBBs. *INAP dependent GSBB*, *INAP independent GSBBs* and *pictorial GSBBs*. The first class, the *INAP dependent GSBBs* translate into a sequence of B-INAP messages that are exchanged between the B-SCP and B-SSP and/or B-IP. INAP independent GSBBs perform operations that do not relate to INAP operations. The majority relate to database manipulation but also include variable manipulation and service logic flow control. The pictorial class only impacts

the layout of the service picture.

The approach within the broadband SCE created was to reuse the existing SLEE within the narrowband product, but develop new INAP dependent GSBBs to handle the functions required by the B-INAP plus new infrastructure ones to support the broadband functionality such as the abstract view introduced by the Switching State Model and the finite state machine of the service logic.

3.1 Generic Service Building Blocks for broadband services

GSBBs that depend on the new B-INAP, or *B-INAP dependent* GSBBs together with new GSBBs to take into account the impact of the broadband services have been developed. Significantly, B-INAP dependent GSBBs in the original product were retained and are successfully reused without changes. The B-INAP dependent GSBBs developed fall into the following categories:

- Call Control GSBBs
- B-SRF interaction GSBBs
- User-Service Interaction GSBBs
- Other miscellaneous B-INAP dependent GSBBs

The call control GSBBs perform all the necessary operations that allow the connections to be established and deleted between users in the IN service session. The B-SRF interaction GSBBs handle all the operations that instruct the B-SRF to run script based interaction with the user as well as collecting information from the user. The User

Interaction GSBBs allow the service logic to handle the end-to-end protocol between the user and the service logic, where the service creator can construct and extract messages to and from the user. Other GSBBs developed allow the triggering and correct termination of IN services and GSBBs that allow the collection of buffered SSM events as a result of the finite state machine implemented for the B-SCP.

For the call control GSBBs, instead of providing a complete view of the SSM, the service creator is only aware of parties and bearer related/bearer unrelated connections. The service creator uses the dictionary number to reference the party objects and uses the identifiers to identify bearers and has no actual visibility of leg objects. The service infrastructure handles the complex association of objects and identifiers performing the necessary mapping into the SSM model. This has the advantage that the service creator needs not be concerned with the complex object associations and relationships that exist.

Figure 7 provides a list of possible GSBBs. All GSBBs relating to database access were retained without any modifications. These data access GSBBs allow for example the look up for time of day routing or area of origin determination. As there is no impact of a broadband network, the building blocks are unaffected. A GSBB that allows the service creator to define a customised data access caters for any special needs of the broadband services.

3.2 GSBB message sequence

Each of the B-INAP dependent GSBBs translates into a sequence of messages that are exchanged between the B-SCP and the B-SSP and/or B-IP. This message sequence is

translated from a dynamic description of the GSBB where the information in the messages is populated for a combination of the values inserted by the service creator from a graphical user interface, the SSM model and the service infrastructure. The GSBBs have been created to reflect the type of B-INAP operation. Again, verbs such as *Add and Delete* have been applied to describe the permissible actions in the GSBB. However such actions are restricted to connections or real end users. The action of the GSBB will result in either the creation or deletion of all the necessary SSM objects in the B-SSP. For example one of the new GSBBs developed, the *BearerControl* GSBB allows the selection of either the deletion of an existing bearer connection or the creation of a new bearer connection between two existing parties. These actions will map on the *DeleteBearer* or *AddBearerToSession* B-INAP operations. When a new bearer connection is requested, new objects relating to the bearer connection and its associated legs are created. Within the B-SCE this is done automatically by the infrastructure thus removing this task from the service creator. This ensures that consistent SSM information is inserted in the messages to the B-SSP. The service creator needs to supply the bearer characteristics (e.g. bandwidth, class, etc.) and the directory number of the parties that are to share the bearer connection.

Each GSBB has a specific message sequence description that describes the messages that are exchanged between the B-SCP and the B-SSP and/or the B-IP. The replies from the B-SSP take the form of state changes relating to objects. The communication indicating this state change will only be sent if the service logic had previously sent a specific message requesting to be informed about the state changes. Part of the GSBB's function is to arm all necessary state changes in the SSM model to allow the GSBB to

be informed of the outcome of the request of the required operation. Each call control GSBB automatically arms the necessary state change requests in the form of *RequestReportSSMChange* so that it will always be notified about the outcome of a call control request.

5

Figure 5 shows the message sequence exchanged between the service logic and the B-SSP when the *AddBearer* option in the *BearerControl* GSBB is selected. Note, however this shows the expected message sequence under normal conditions and excludes the possibilities of exceptions occurring. The three *RequestsReportSSMChanges* individually arm the state changes that would indicate whether the bearer has been set-up, or either of the end parties has refused to accept the call, or either of the parties has abandoned the call. The GSBB has a status field that will be populated with a value that indicates the outcome of the *AddBearer* request. Additional GSBBs are available to allow the service creator to test the possible values of the status field and take appropriate action in the service picture.

10

15

20

In order to simplify the service creation process, the GSBBs will only allow sequential operations to be performed. That is, until a notification of outcome of the operation is reported, the GSBB will not allow another operation to be performed. As sequential operations are performed, it is possible that a state change is reported that does not relate to the operation that the GSBB has been asked to perform, i.e. for example the release of a separate bearer connection within the same IN session. In such a case the normal message sequence is interrupted to allow this message to be read and its content buffered as an asynchronous event. This is evident in the run-time flow description of the GSBB

in figures 6A and 6B. Once the event is buffered, the GSBB will resume its normal action as dictated by the dynamic description. Additional GSBBs have been provided to test and read for such buffered events to allow them to be handled as dictated by the service requirements.

5

3.3 GSBB run-time action

In addition to a message sequence file, each GSBB has an associated *run-time action file* that describes the actions that the icon will perform. The actions cannot be altered by the service creator once the appropriate GSBB operation has been selected. Figure 6 shows the run-time actions for the AddBearer option in the BearerControl GSBB as described by the message sequence in figure 5.

10

3.4 Service Creation Environment Architecture

The central architecture of the B-SCE is shown in figure 8. The architecture has three distinct domains: *building block definition*, picture editor and *code generation*.

15

The *building block definition* is the stage where the generic service building blocks themselves are defined. These include the definition of the message sequence chart described in section 3.2, the runtime actions described in section 3.3. The separation between this definition and the graphical picture editor means that future capabilities are not constrained by the graphical editor.

20

The *picture editor* is a graphical tool that allows the service creator to select the building blocks from a palette and drag them to a drawing area. They can then be linked together

to form a logic flow required by the service requirements. Each building block consists of a property sheet that contains input fields, menus and selection buttons are used to specify the detailed operation. Such a detailed operation may relate to a decision branch criteria, user interaction information to be sent to the end user or the bearer characteristics required to set up a particular bearer connection. The output of the picture editor is a general logic description of a service, without any explicit reference to the B-INAP other than those at a higher level.

This picture file is acted upon by the *code generator* to create runtime files required to execute the service on a B-SCP. As the picture files do not contain any information about the messages, the code generation uses the GSBB message sequence details and runtime actions to perform the necessary translations for the B-INAP operations. In addition, SSM details and the asynchronous message library details are also used by the code generator to produce the ANSI C source code for the Service Logic Program.

4. Conclusion

The application of an Intelligent layer to broadband networks can be facilitated by a graphical service creation environment. A flexible narrowband system can be expanded to provide the additional complexity imposed by this environment. To realise the services utilising this environment such as advanced broadband video-conferencing the separated control of connections is essential, together with the asynchronous handling of events.

In addition service logic will be required to interwork with complex nodes such as Intelligent Peripherals to provide the User with comparable service navigation techniques.

CLAIMS

1. A broadband graphical service creation environment for creating Intelligent Network services for an Integrated Broadband Service Digital Network/Intelligent Network network, comprising building blocks, a picture editor and code generator, the building blocks being a collection of broadband specific actions, data access and data manipulation routines plus pictorial blocks that define the graphical layout of the service.
2. A broadband graphical service creation environment as claimed in Claim 1, comprising B-IN Application Protocol dependent Generic Service Building Blocks to handle functions required by a B-IN Application Protocol.
3. A broadband service creation environment as claimed in Claim 1 and further comprising Generic Service Building Blocks capable of running specific message sequence, which sequence can be interrupted by some asynchronous events, wherein once the event is handled, the original message sequence can be resumed and the service creation environment is able to generate Service Logic Programs (SLPs) that can support a finite state machine.
4. A broadband graphical service creation environment substantially as hereinbefore described, with reference to and as illustrated in the accompanying drawings.

This Page Blank (uspto)

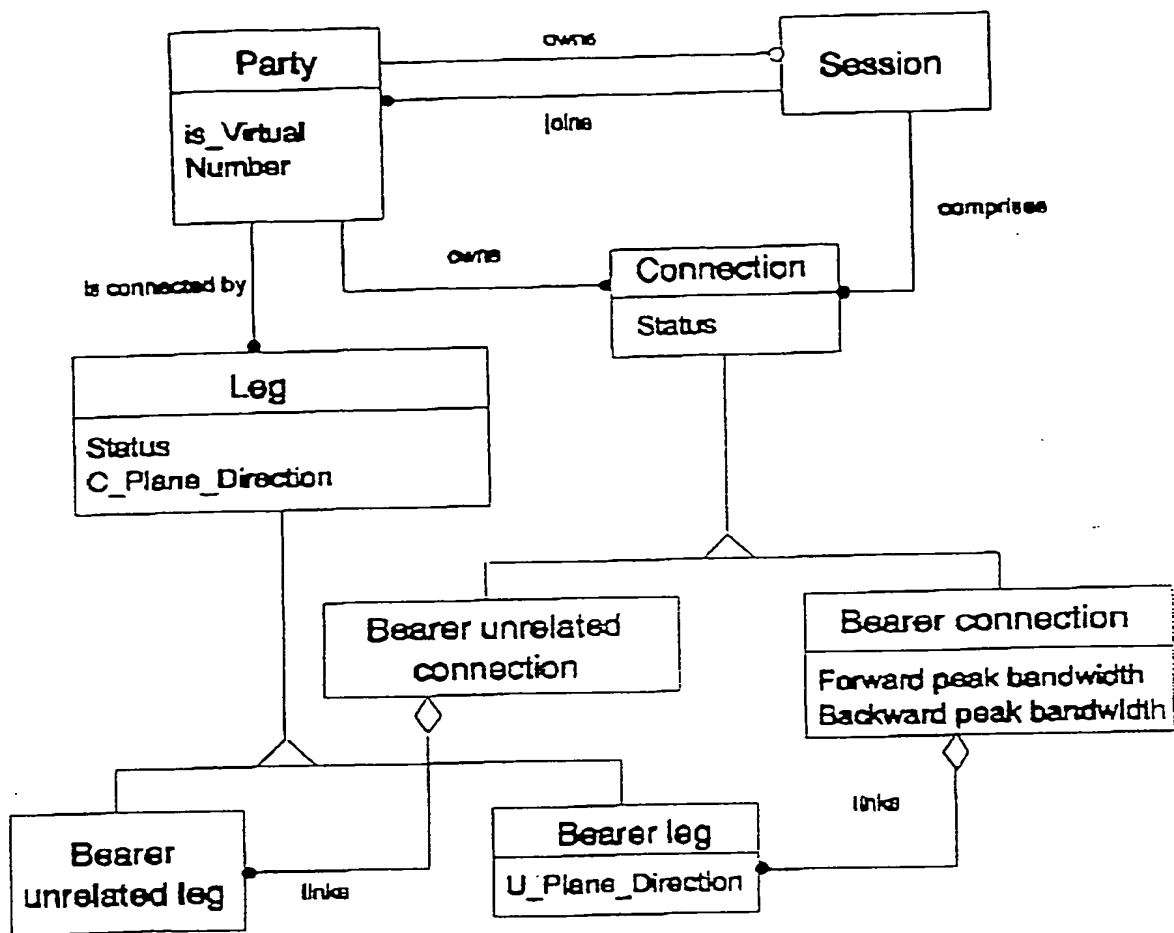


FIGURE 1

This Page Blank (uspto)

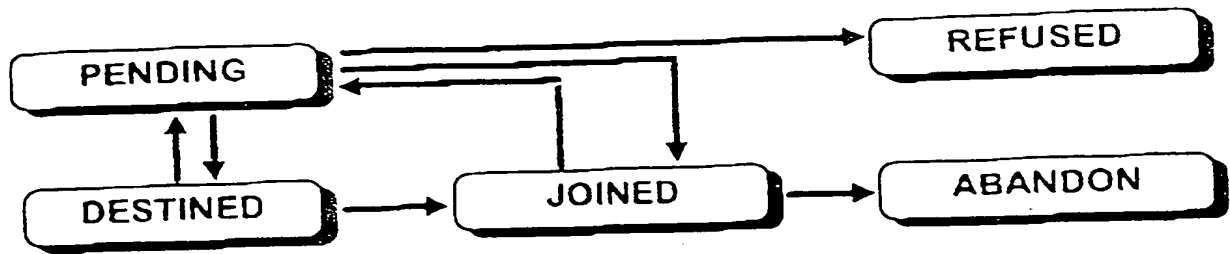


FIGURE 2

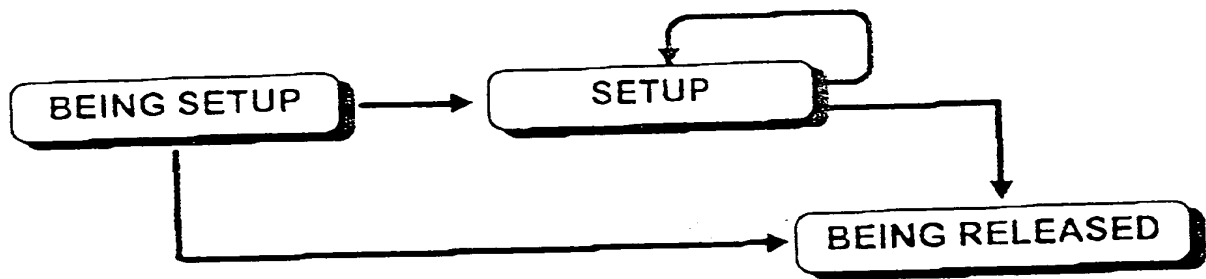


FIGURE 3

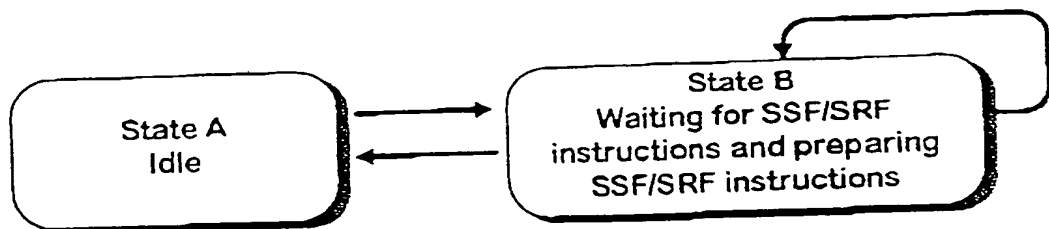


FIGURE 4

This Page Blank (uspto)

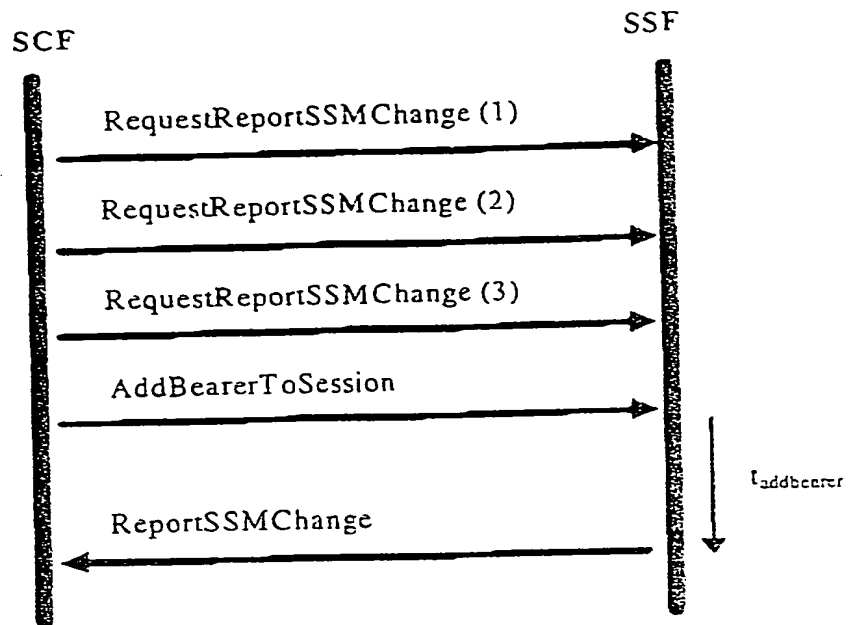


FIGURE 5

This Page Blank (uspto,

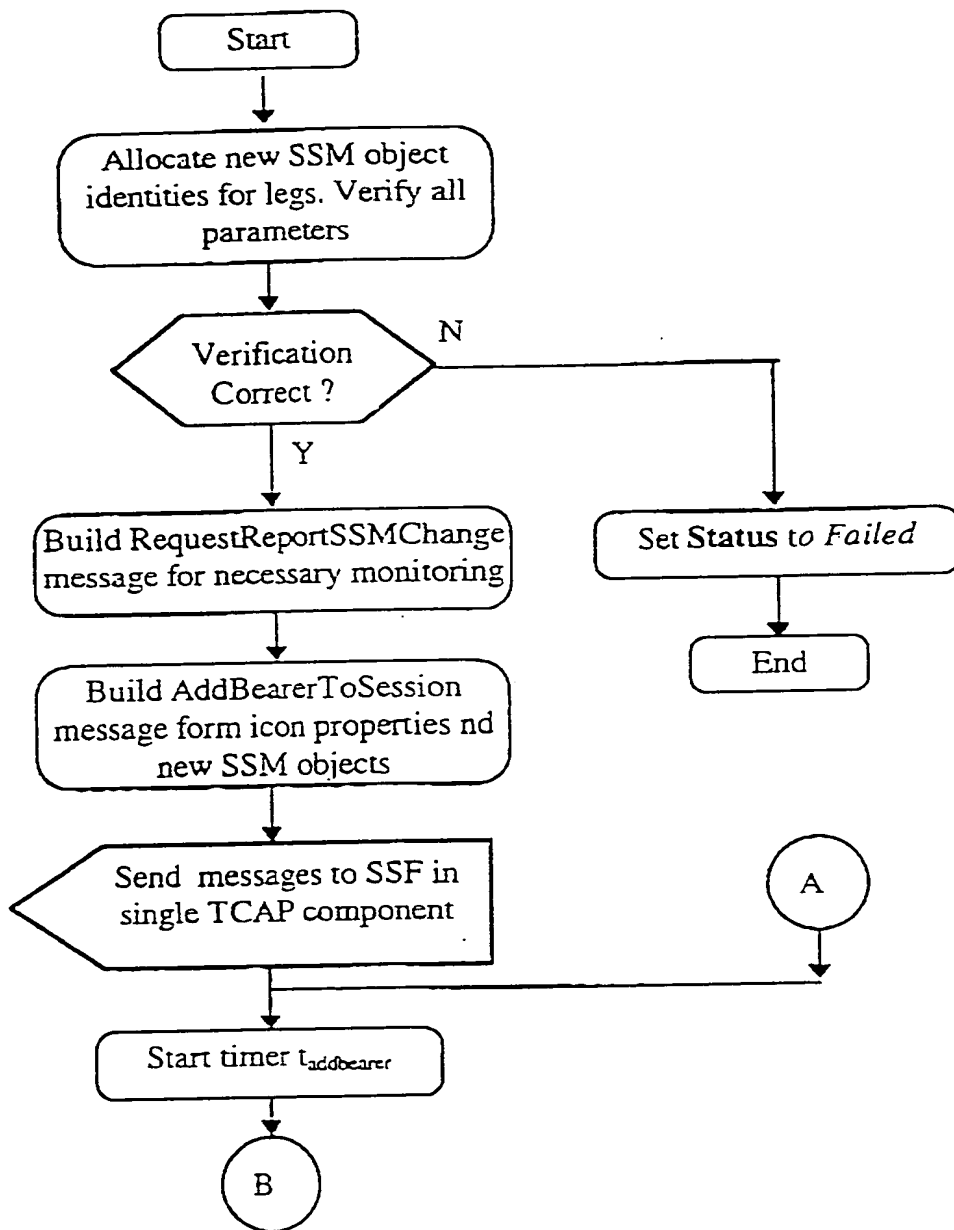


FIGURE 6A

This Page Blank (uspto)

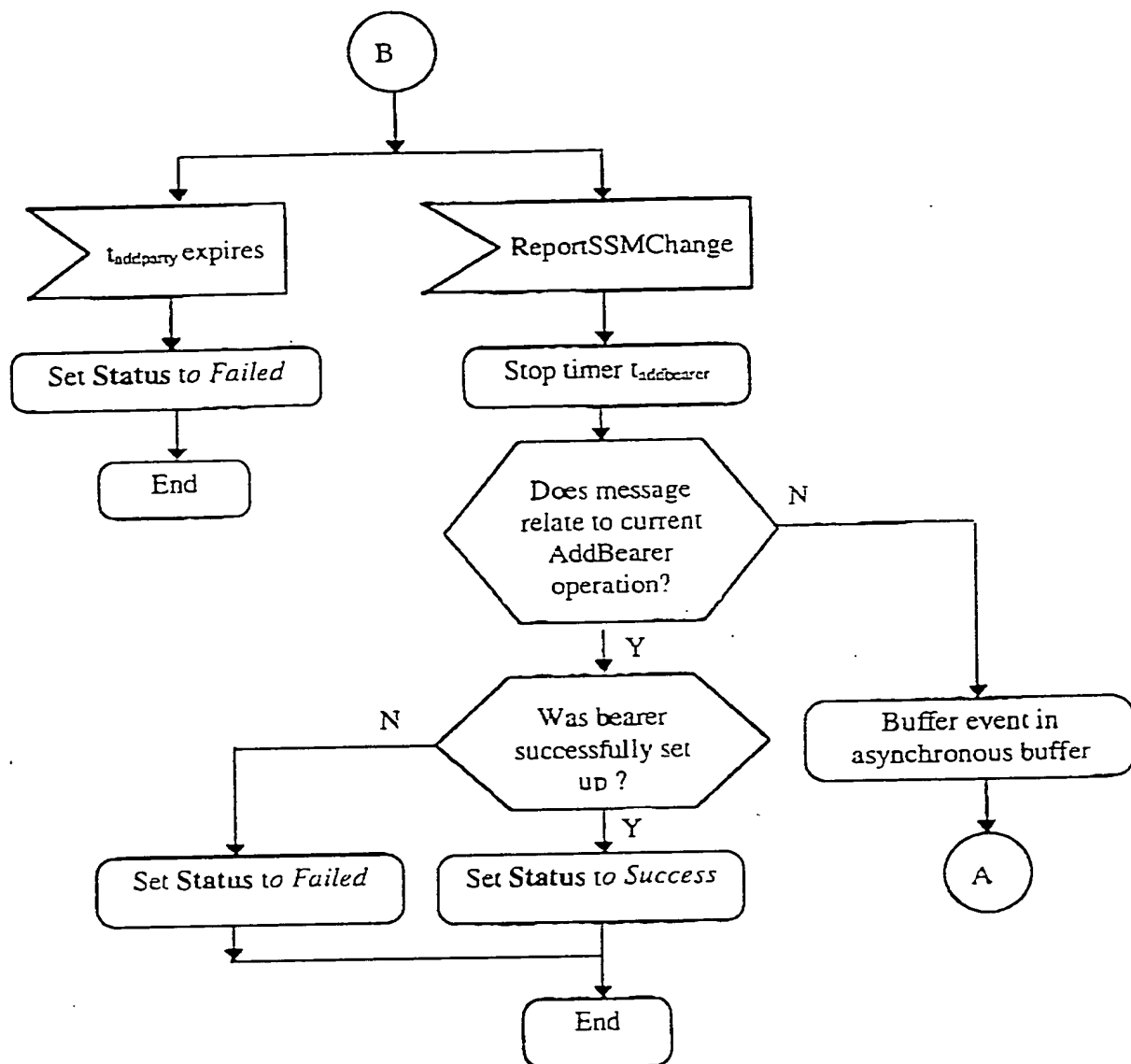


FIGURE 6B

This Page Blank (uspic,

	Summary of Function
Call Control GSBBS <i>BearerControl</i> <i>PartyViewControl</i> <i>JoinParty</i>	<p>Adds or drops bearer connections between existing parties in the session.</p> <p>Adds a new party or drops an existing party in the session.</p> <p>Routes a user initiated connection, or adds a new party to an existing IN session.</p>
<i>IPConnect</i> <i>Continue</i>	<p>Connects a user to an B-IP using various connection options allowed by the B-INAP.</p> <p>Restores call processing after an interrupt. by the service logic</p>
IP Interaction GSBBS <i>PlayCollect</i>	<p>Specifies a script to be played at the B-IP and information is to be collected from the user.</p>
<i>IPPrompt</i> <i>IPMonitor</i>	<p>Specifies that a script is to be played at the B-IP</p> <p>Monitors responses from the B-IP that can contain information from the user</p>
USI GSBBS <i>USISpecify</i> <i>USIElement</i> <i>SendUSI</i>	<p>Allows the service creator to initiate the specification of a USI Component</p> <p>Allows the service creator to specify or extract a single constituent IE of a USI component</p> <p>Allows the service creator to request that the previously specified USI component is sent to the user.</p>
Other GSBBS <i>Start</i> <i>End</i> <i>AsyncRead</i> <i>Monitor</i>	<p>Initial trigger for an IN session</p> <p>Terminates an IN session in a correct and orderly fashion</p> <p>Allows the reading of buffered events such as bearers that have been released.</p> <p>Sets the service logic in a monitor state</p>

FIGURE 7

This Page Blank (uspto)

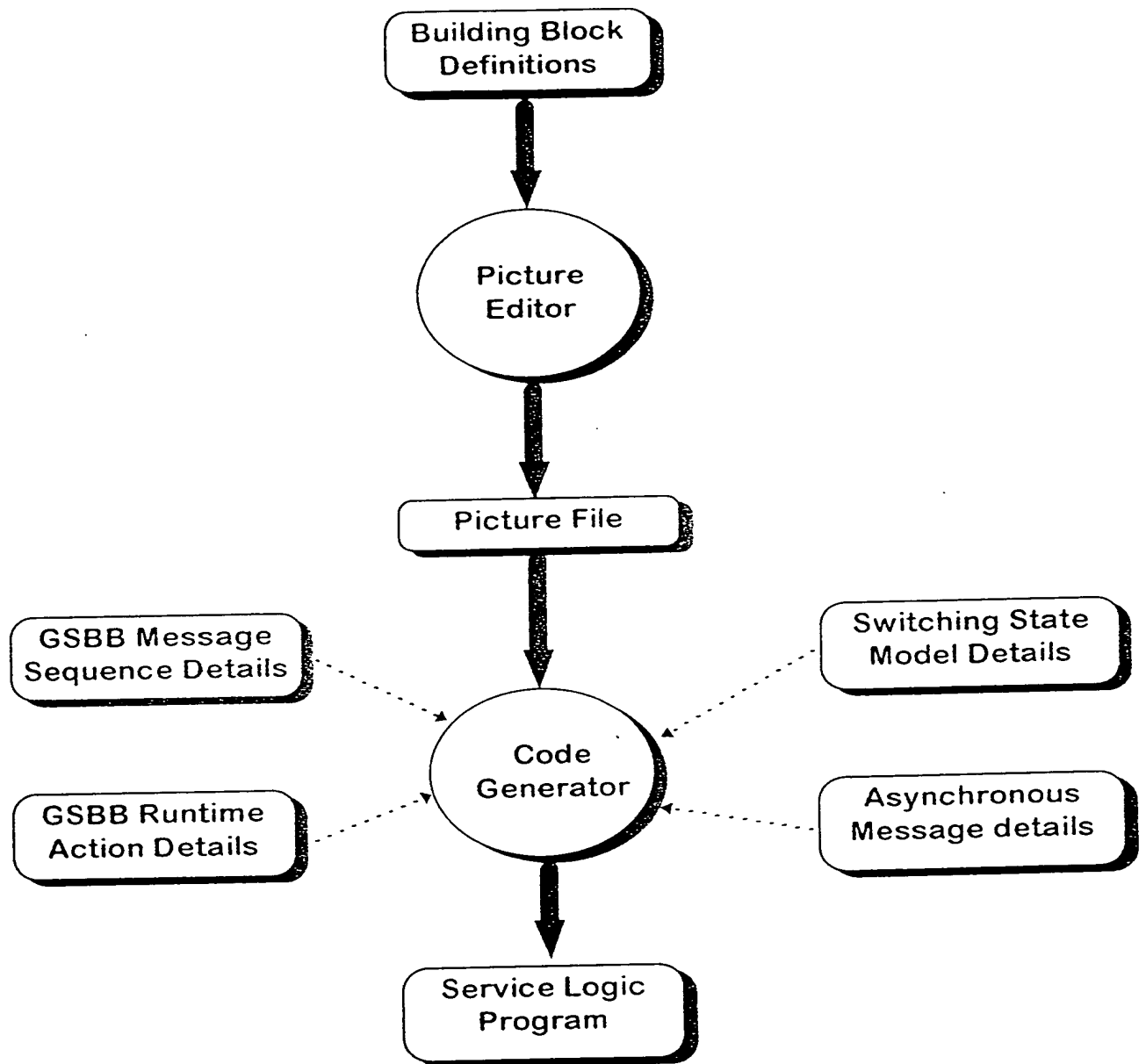


FIGURE 8

PC716-B 44/105/19

9/3/99

Lee Patent Dept

This Page Blank (uspto)